Automated Conversions to IBM DB2 for z/OS

IBM Redbooks Solution Guide

Business agility is essential to client success in the fast-paced, competitive, and highly regulated global business climate of today. To grow and thrive, clients must effectively adopt and apply modern techniques (including analytics) to gain and sustain a competitive edge. An inflexible technology that is not based on standards hinders the ability of a client to meet its business needs.

IBM® DB2® for z/OS® and IBM System z® are an ideal and logical target for migrations from prerelational databases. New and existing clients rely on DB2 for z/OS and System z for their unsurpassed total system reliability, availability, serviceability, and security. Demonstrated benefits include exceptional availability and scalability, a superior level of protection for business-critical data and applications, ease of integration and management across platforms, and reduced infrastructure complexity.

Today the DB2 family of products (see Figure 1) offers an integrated end-to-end solution. It covers all platforms and includes existing data, from which clients can draw large benefits.

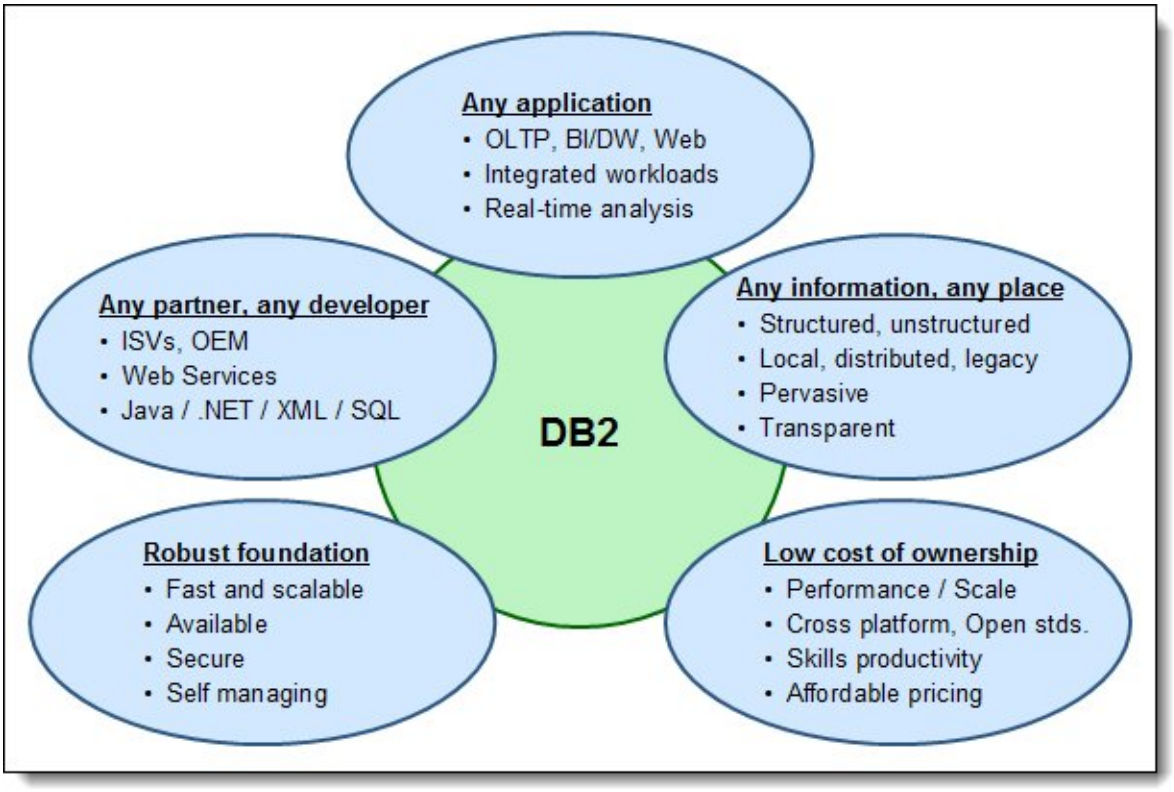

Figure 1. DB2 solutions

Did you know?

In the past, the amount of manual rewrite work needed to migrate a prerelational database and application far exceeded the work that was done by automated tools. However, with experience, innovations, and time, this dynamic changed. The advent of highly automated conversion solutions and methods made prerelational conversions viable and repeatable. This change reduced the amount of time needed, the risks involved, and the costs of the migration process.

Clients use automated conversion solutions to successfully migrate to modern platforms, such as DB2 for z/OS and System z. Several significant migrations of prerelational database to DB2 for z/OS have been completed, and many more are in progress. This trend is expected to continue as maturing conversion technologies become more popular and are widely accepted because of successful migrations.

Business value

Relational databases are the superior data stores of choice for over 25 years. Relational technology is the method of choice for storing large amounts of data with flexibility and easy access to data with a common language, Structured Query Language (SQL). Data stored in tables is accessible and is manipulated by SQL. This data accessibility, when combined with a language suitable for users, results in reduced maintenance and greater productivity.

Today's IT shops are complicated. You likely support multiple types of hardware and software solutions. Although mean time between failures (MTBF) continues to improve, numerous components present the opportunity for a single point of failure.

DB2 for z/OS and the System z platform offer the foundation of resiliency for your environment, keeping your IT assets available to your internal and external customers. DB2 for z/OS and the System z platform provide the architecture to maintain maximum availability, when problems occur, and to adjust to changing and unpredictable workloads.

DB2 is integrated into the System z operating system (z/OS) software and hardware. Through this integration, DB2 can fully use the performance features of the underlying hardware and software. As such, DB2 delivers a level of performance that is unmatched by other mainframe-based relational database management system (RDBMS) products. Because mainframe-based RDBMS products must support work that originates from transaction processing monitors and from distributed systems (such as a client/server), the RDBMS must provide well-defined, efficient interfaces to those systems to facilitate application performance. DB2 provides these interfaces that support both mainframe transaction processing monitors, such as IBM CICS® with the DB2 Adapter from IBM, and client/server systems, with support for stored procedures. These interfaces are well-designed and efficient, and they distinguish DB2 as the enterprise server of choice.

Solution overview

As companies benefited from automation, they developed many applications that use customized software to run their business. Most of these applications use 3270 screen interfaces, large overnight batch jobs, and DBMS or flat files that are prerelational in terms of how data is organized and accessed. With the advent of browser interfaces, RDBMS, and service-oriented architectures (SOA) that support more flexible access to information, many new development options are available.

To take advantage of these development options, companies are looking at alternatives to convert from their prerelational DBMS to a relational DBMS such as DB2 for z/OS. A manual conversion is one alternative, but most companies choose an approach that uses the automated conversion tools. This approach offers the following benefits:

- Application source code maintenance is easier because only the call structure is changed.
- Employees are familiar with converted programs.
- The user interface often remains unchanged.
- The conversion process is completed quickly and with less risk.
- The results of the conversion are easier to test and verify because application functionality is the same.
- The skills and expertise of client personnel are used.
- Changes to data and program structures are standardized.
- Program and database changes during the conversion are accommodated.
- The conversion process is documented in a before log and after log.
- Most automated conversion tools scan the program source code and database structures to generate reports. The reports are used to understand the application relationships for conversion decisions, identify missing and duplicate programs, and identify program interfaces for testing.

An automated conversion requires selection of a vendor to provide the tools to convert the data and the applications. IBM partners with several vendors to assist companies with conversion from prerelational databases to DB2 for z/OS. After a vendor is identified, database conversions typically involve four stages, as shown in Figure 2.

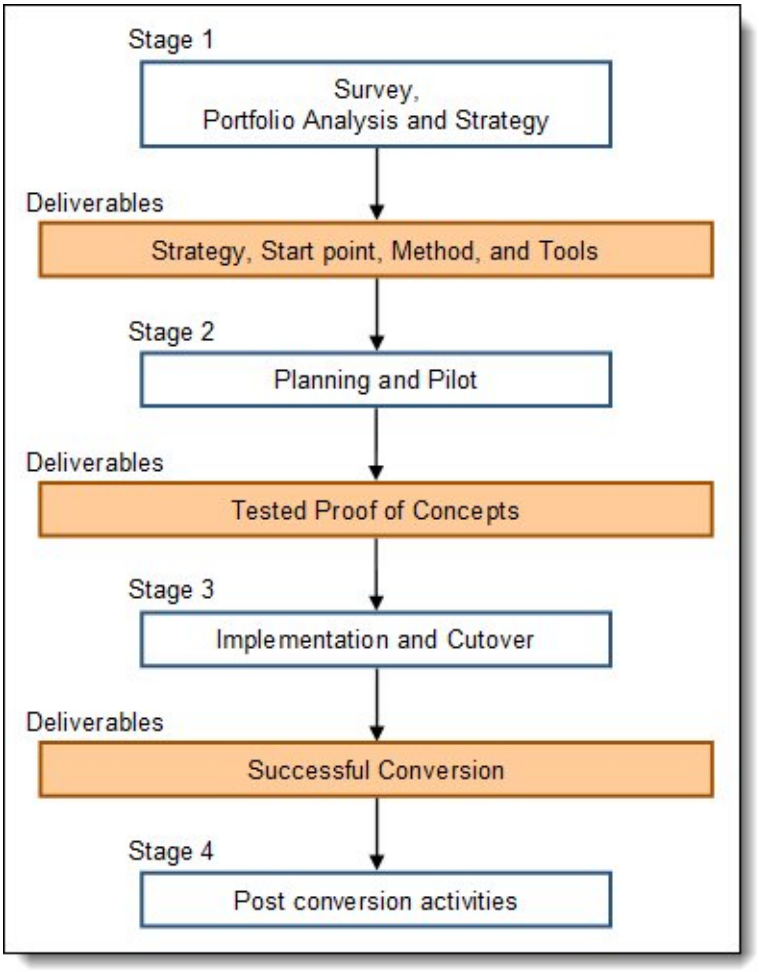

Figure 2. The four stages of conversion

Each stage has the following primary objectives:

- The *assessment stage* involves analysis of your existing environment, including the database and the applications, to arrive at an estimate for the conversion effort to determine the initial project scope and budget.
- The *pilot stage* identifies a small group of transactions, programs, and data. It uses those components to perform a small conversion that acts as a feasibility study and provides learning points for the complete conversion.
- The *conversion stage* is the culmination of the work that was completed in stage 1 and stage 2. Project planning, coexistence, testing, change control, database design, code conversion, and manual rework are done to complete the conversion to the new database.
- The *post-conversion activities stage* allows companies to perform more activities upon the converted database. It also allows applications to take advantage of further database tuning, more usage of relational functionality, a more relational data model, application rearchitecture, and usage of database tools.

Solution architecture

The automated conversion approach uses the customer source code, data definitions, and data as input to a conversion tool. The tool then applies conversion rules that are provided by the customer during the conversion stages. The tool generates the converted source code and data and then, generates the DB2 data definition language (DDL) statements that are required to create the DB2 database. Figure 3 illustrates this process.

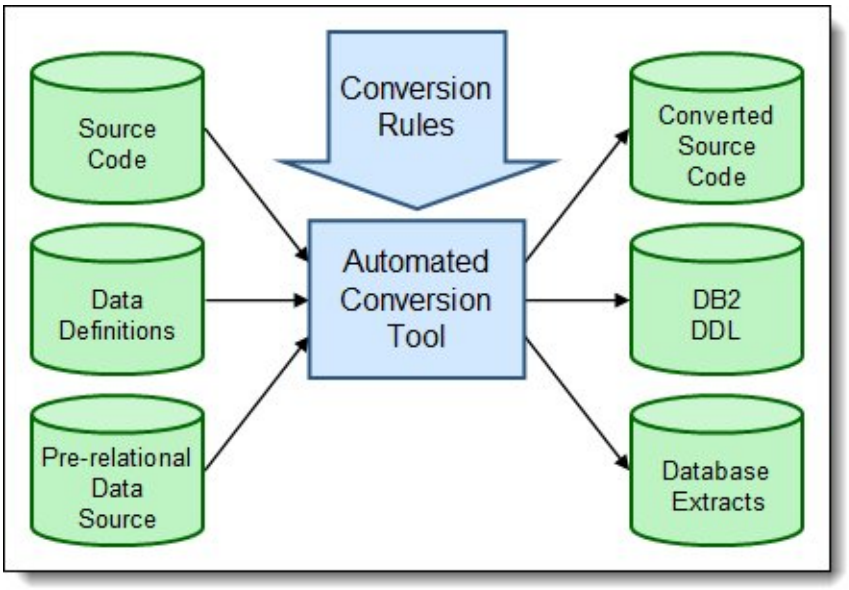

Figure 3. Solution architecture

The automated conversion approach requires a thorough understanding of the existing data and applications. In some cases, the format of the data in the prerelational database does not always fit into an equivalent DB2 data format. Dates and times are an example. By using DB2, you can store data in a date and time format, with strict requirements that the data must conform to valid dates and times. The prerelational database might not have such strict requirements. In these cases, you must define conversion rules and provide them as input to the conversion tool.

The conversion tool applies all the rules that are supplied to arrive at the converted source code, DDL statements, and converted data for loading into DB2. The customer then uses their DB2 environment to create the database (by running the DDL statements), to load the database (by using the database extracts), and to prepare the converted source code.

Tests of the conversion process might identify issues in any of the three components that are converted. After each test, the customer must modify the conversion rules to correct any issues and then rerun the conversion process. This testing process is likely to involve multiple iterations of rule definition and subsequent testing before all the conversion rules are correct for the data and applications that are being converted.

Usage scenarios

Customers choose to convert to DB2 for z/OS most often for the following reasons:

- Users or external customers demand access to relational data.
- They want to simplify the number of supported database platforms.
- They lack available customer staff who are skilled in DBMS.
- Application agility is needed to meet changing business and technical demands that require frequent programming and data model enhancements.

DB2 for z/OS provides a robust database and supporting tools to address the reasons that customers convert from their prerelational databases as the following example demonstrates.

A large government institution was faced with the need to modernize its IT infrastructure. This customer's Integrated Database Management System (IDMS) application was converted from VSAM in the 1980s. As the customer base increased and data access requirements expanded, becoming more complex, changing the application and supporting database to meet regulatory compliance became more challenging. Changes often included the use of the same field for multiple purposes or the use of filler fields to hold data. In addition, support for the installed version of the IDMS software was scheduled to end within a few years, and the customer did not want to pay for both DB2 and IDMS license and support charges.

In late 2007, the customer initiated a project to convert from IDMS to DB2 for z/OS and to evaluate automation tool vendors and conversion support services. The sheer size of the conversion effort dictated that an automation conversion approach for the project must be found. The following statistics of the project reinforced the need for an automated approach:

- Approximately 7 million lines of code
- Over 1,400 batch programs
- Over 400 CICS programs
- Almost 4 terabytes of data

The customer selected a conversion vendor and contracted with IBM Information Management (IM) Lab Services to provide database consulting services throughout the conversion effort. IM Lab Services was involved in the database design, application design, testing, implementation, and post-production tuning efforts. IM Lab Services also provided customized DB2 workshops with hands-on labs for database administrators and application developers.

After the customer selected its conversion vendor and IBM services were assigned to the project, the conversion project started. The project consisted of the following stages:

- Identifying and conducting a pilot project
  - The customer identified a self-contained application to be part of the pilot, and the customer and the conversion vendor agreed to a contract to conduct a pilot project. The following goals of the pilot project were identified:
    - Demonstrate the capabilities of the conversion tool.
    - Verify the compatibility of the tool with the environment of the customer.
    - Identify problem areas in preparation for the larger conversion effort.
  - This experience and knowledge was used throughout the subsequent steps of the larger conversion effort.
- Automated assessment

  A full analysis of all application and database components was performed to estimate the scope of the conversion. This process was conducted iteratively because some components were identified as missing whenever the assessment was run. The components were missing because some source code or jobs that were found in nonstandard libraries were not provided to the conversion vendor. The output of this step included an estimate of the scope of the conversion and reports that identified the components to be converted. It also included a project plan that detailed the steps of the project and the process to collect the components targeted for conversion.
- Collecting, converting, and customizing the application and database
  - The conversion vendor ran jobs to collect the IDMS components necessary for the conversion: source code, copybooks, job control language (JCL), CICS maps, database definitions, and data. All IDMS components were processed by the automated conversion tool to produce the equivalent DB2 components. The new components were then sent to the customer.
  - At the beginning of this production step, certain business rules were predefined as input to the conversion process. These rules allowed for renaming fields to conform to DB2 standards and for defining the manner in which differences in IDMS and DB2 data types were managed.
  - In addition, some customization was conducted to determine which buffer pools to use and to partition certain tables. After the rules were defined and the converted applications and database were returned, the DB2 objects were created.
- Application and database testing
  - In this stage, application and data conversion testing were conducted simultaneously. For these tests, small amounts of data were loaded into unit test environments to allow application testers to begin their analysis. At the same time, tests of DB2 LOAD utilities were run on the completed set of data to identify any data conversion errors or DDL errors.
  - The testing continued through multiple iterations. Testing progressed through unit test, system test, and performance test phases. All defects were logged in to a defect tracking system, with each defect identified by a defect number. Regular weekly meetings were held with the conversion vendor to review the testing results and any defects that were open.
- Performance testing
  - Throughout the testing phases, applications that were potential performance bottlenecks were identified. Individual teams were organized to address online performance issues and batch performance issues.
  - The online performance testing identified a mix of transactions that were deemed critical for online performance. Automated testing software was used to capture the transaction mix and rerun the mix as needed after program or database configuration changes were made.
  - The batch testing identified the longest running batch jobs within IDMS. Equivalent batch jobs that were run within DB2 also were identified and the difference between the two sets of batch jobs was measured. Decisions were then made about which programs must be rewritten and which jobs could be addressed by database tuning.

- Implementation tests
  - Several implementation dry-run tests were conducted to validate the production implementation process. The tests also provided an estimate of the amount of time that is needed to complete the production implementation. The following tasks were included in the tests:
    - Running the entire data extract and load process
    - Running the processes to switch from the IDMS application load libraries to the DB2 application load libraries
    - Testing the new code with the converted data
    - Backing out the load libraries to point the application back to the IDMS database
  - These implementation tests required considerable planning. The tests included all of the conversion process personnel, including application developers, database administrators, CICS administrators, testers, users, operators, security administrators, change management staff, and project management staff.
  - The lessons learned from each test were documented and changes to the process were applied to next steps. These changes minimized the chance for errors during production implementation.
- Production implementation
  - The production implementation was completed during a holiday weekend. The same steps that were conducted during the implementation tests also were conducted in the production environment. Initial validation testing was conducted to ensure that there were no issues and no need to roll back to the IDMS version of the applications and data. After the validation testing was completed and a decision was made to start production, batch processing began.
  - Teams were identified to monitor both the batch and online workload during the first few weeks of the processing. The conversion vendor also provided onsite support for most of the first week after the implementation was complete. The few problems that were found were addressed quickly. Most problems occurred when JCL was translated between the test environment and the production environment.
- Post-conversion tuning
  - Performance data that was captured for the first few online and batch cycles was reviewed to identify areas of concern. The online transactions and batch jobs were completed within the accepted service level agreements (SLAs) for elapsed time. However, the processor cost for most jobs was considerably higher than the cost under IDMS. The conversion was same-to-same. Therefore, an increase in the processor cost was expected because the access of the DB2 data was left similar to how IDMS accesses data.
  - Changes were made to take advantage of SQL functionality by replacing programmatic joins with SQL joins and by including filtering logic in the SQL instead of in the program.
- Relational redesign
  - After the conversion was complete and was stable in production for a few months, an effort began to determine how the application and database might take advantage of true relational database capabilities. This analysis step is a long-term project that involves the following tasks:
    - Reduce dependency on generated I/O modules
    - Take advantage of SQL capabilities
    - Remove any IDMS-type constructs from the database design
  - The final goal of the redesign is to have an application and database that are truly relational. The time and effort to incorporate this change as part of the first phase unnecessarily lengthens the project. The effort to make these changes up front also increases the dependency on IDMS beyond the end of the support date for the installed version of IDMS. Therefore, a decision was made to use an automated conversion solution. The application and database must be reviewed after the conversion, however, and plans must be made for a true relational implementation.

IM Lab Services was involved throughout every stage of the conversion process and is still involved today to provide guidance for how the customer can implement a relational redesign of the converted database and supporting applications.

Integration

The close relationship between DB2, z/OS, and the System z platform is unique in the industry. It delivers world-class performance, scalability, and availability, which benefits all sizes of businesses including the top companies in the world today that run their important applications. This tight partnership is a strategic ongoing IBM goal and a significant differentiator.

Supported platforms

IBM DB2 for z/OS runs on the z/OS operating system on the System z platform. Data that resides on DB2 for z/OS can be accessed from just about any known operating system and platform through the IBM Distributed Relational Database Architecture™ (IBM DRDA®) and requests to DB2 that are not DRDA. DRDA is used for transparent access to distributed data and increased manageability of distributed data across heterogeneous DBMS systems.

Ordering information

The Migration Practice of IBM nurtured relationships with skilled business partners over many years. These partners are not new to IBM. Instead, they are conversion vendors with whom we have developed long-term relationships and worked with successfully.

For information about worldwide solutions from IBM business partners, see this website:
http://www.ibm.com/partnerworld/gsd/homepage.do

For information about the IBM DB2 Migration Project Office, see this website:
http://www.ibm.com/software/solutions/softwaremigration/dbmigteam.html

Related information

For more information, see the following documents:

- *Streamline Business with Consolidation and Conversion to DB2 for z/OS*, SG24-8044
  http://www.redbooks.ibm.com/abstracts/sg248044.html
- IBM Offering Information page (to search on announcement letters, sales manuals, or both):
  http://www.ibm.com/common/ssi/index.wss?request_locale=en

  On this page, enter `DB2 for z/OS`, select the information type, and then click **Search**. On the next page, narrow your search results by geography and language.
- DB2 for z/OS product page
  http://www.ibm.com/software/data/db2/zos

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service. IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you. This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products. This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurement may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

This document was created or updated on April 5, 2013.

Send us your comments in one of the following ways:

- Use the online **Contact us** review form found at:
  ibm.com/redbooks
- Send your comments in an e-mail to:
  redbook@us.ibm.com
- Mail your comments to:
  IBM Corporation, International Technical Support Organization
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400 U.S.A.

This document is available online at http://www.ibm.com/redbooks/abstracts/tips0967.html .

Trademarks